Dakota Software Training

Dakota Overview

http://dakota.sandia.gov

Exceptional

service

in the

national

interest

Module Learning Goals

- What is Dakota?

- Why use Dakota?

- Prerequisites for Using Dakota

- Training outline

WHAT IS DAKOTA?

Dakota enhances simulations…

Algorithms for design exploration and simulation credibility
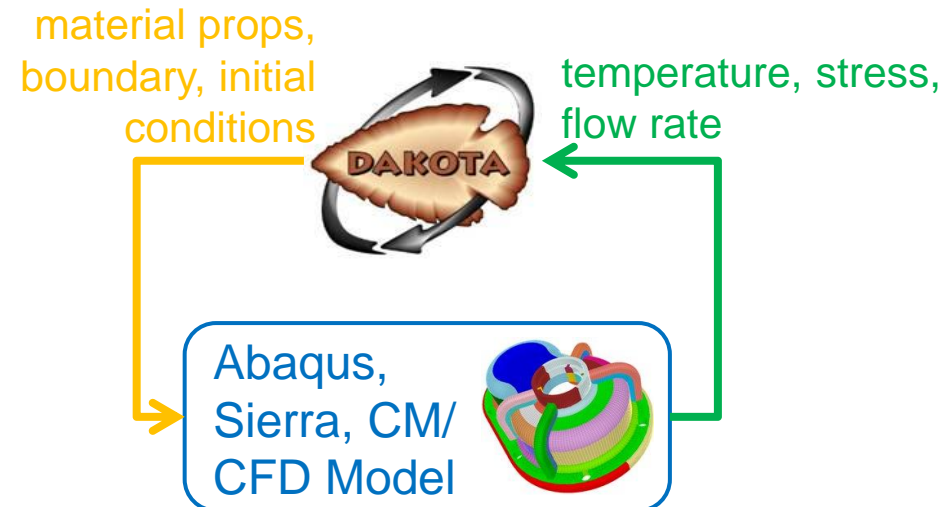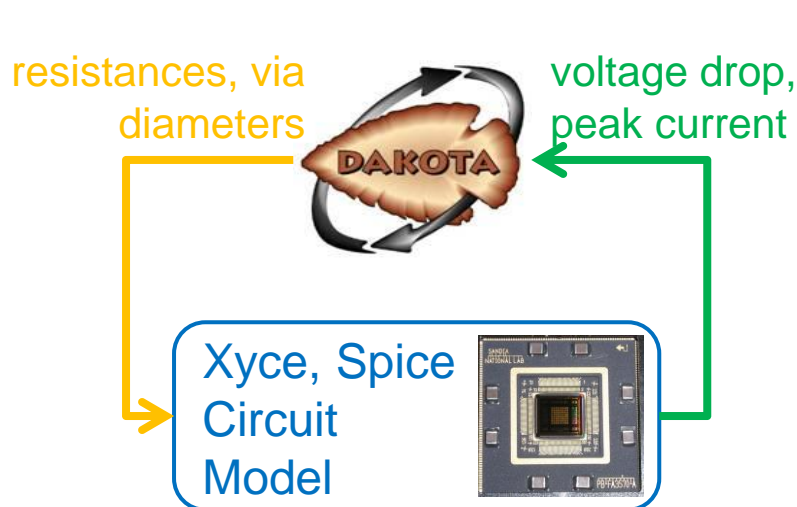
- Suite of iterative mathematical and statistical methods that interface to computational models

- Makes sophisticated parametric exploration of simulations practical for a computational design-analyze-test cycle

- Provides scientists and engineers (analysts, designers, decision makers) greater perspective on model predictions:
  - *Enhances understanding of risk by quantifying margins/uncertainties*
  - *Improves products through simulation-based design, calibration*
  - *Assesses simulation credibility through verification and validation*

…by analyzing ensembles

- Strategically selects model parameters
- Manages concurrent simulations
- Analyzes responses (model outputs)
- Automates one-pass parameter variation/ analysis to advanced goal-oriented studies

Run	Input	Output
1	0.814	91.3
2	0.906	63.24
…		
N	1.270	9.75

resistances, via diameters → DAKOTA → voltage drop, peak current → Xyce, Spice Circuit Model

material props, boundary, initial conditions → DAKOTA → temperature, stress, flow rate → Abaqus, Sierra, CM/ CFD Model

Key Questions Answered

Dakota makes iterative parametric analysis practical for black-box simulations to answer questions of:

- Sensitivity: Which are the crucial factors/parameters?
- Uncertainty: How safe, reliable, or robust is my system?
- Optimization: What is the best performing design or control?
- Calibration: What models and parameters best match data?

Indirectly:

- Verification: Is the model implemented correctly, converging as expected?
- Validation: How does the model compare to experimental data, including uncertainties?

Enables quantification of margins and uncertainty (QMU) and design with simulations; analogous to experiment-based QMU and physical design/test.

Sensitivity Analysis

- ***Which are the most influential parameters?***

- Interrogate model to assess input/output mapping
  - Expose model characteristics, trends, robustness
  - Focus resources for data gathering or model/code development
  - Screening: reduce variables for UQ or optimization analysis

- ➤ Dakota automates common single parameter variations, and provides richer global sensitivity methods
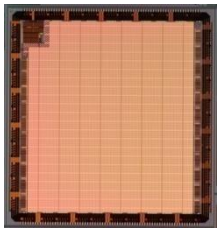
- Xyce model of CMOS7 ViArray
- Assess influence of manufacturing variability on supply voltage performance during photocurrent event
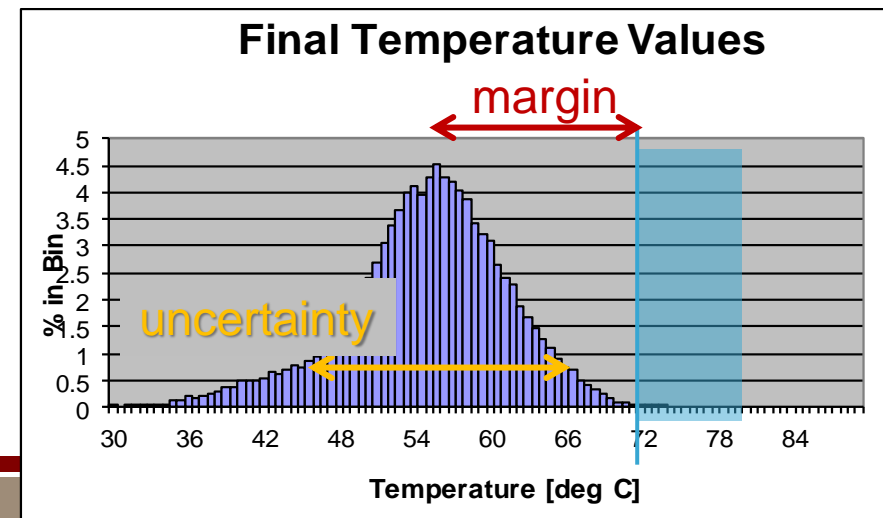

partial correlation
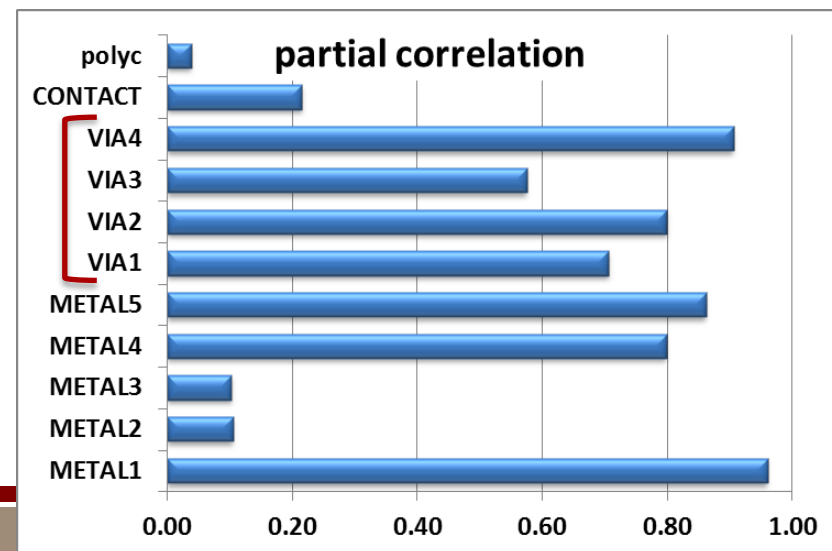
Uncertainty Quantification

- ***Given parameter uncertainty, what is the uncertainty in the model output?***
  - Mean or median performance of a system
  - Overall variability in model response
  - Probability of reaching failure/success (reliability)
  - Range/intervals of possible outcomes
- ➢ UQ also enables statistical validation metrics
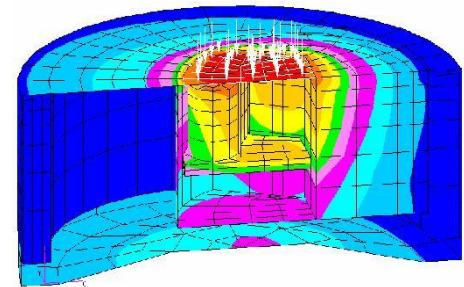
- Device subject to heating, e.g., modeled with heat transfer code
- Uncertainty in composition/ environment (thermal conductivity, density, boundary)
- Make risk-informed decisions about margin to critical temperature

Final Temperature Values

margin

uncertainty

% in Bin: 5, 4.5, 4, 3.5, 3, 2.5, 2, 1.5, 1, 0.5, 0

Temperature [deg C]: 30, 36, 42, 48, 54, 60, 66, 72, 78, 84

Optimization

- ***Goal-oriented: find the best performing design or scenario, subject to constraints***
  - Identify system designs with maximal performance
  - Determine operational settings to achieve goals
  - Minimize cost over system designs/operational settings
  - Identify best/worst case scenarios


fuel tanks

- Computational fluid dynamics code to model F-35 performance
- Find fuel tank shape with constraints to minimize drag, yaw while remaining sufficiently safe and strong

Calibration / Parameter Estimation

- *Data-driven: find parameter values that maximize agreement between simulation output and experiment*
    - Seek agreement with one or more experiments, or high-fidelity model runs
    - Yields: single best set, range, or distribution of parameters most consistent with data
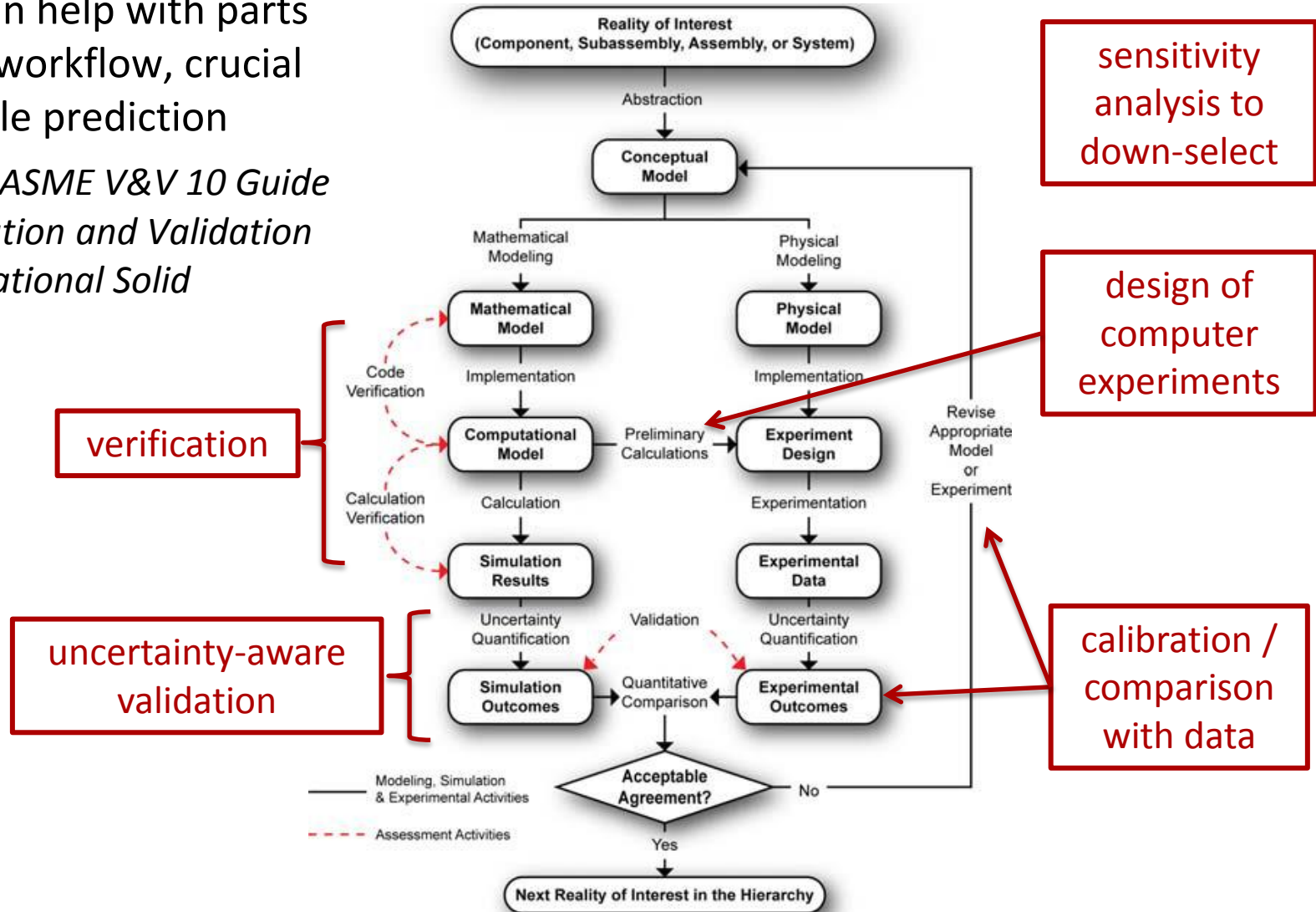
- Calibrate material model parameters to match experimental stress observations
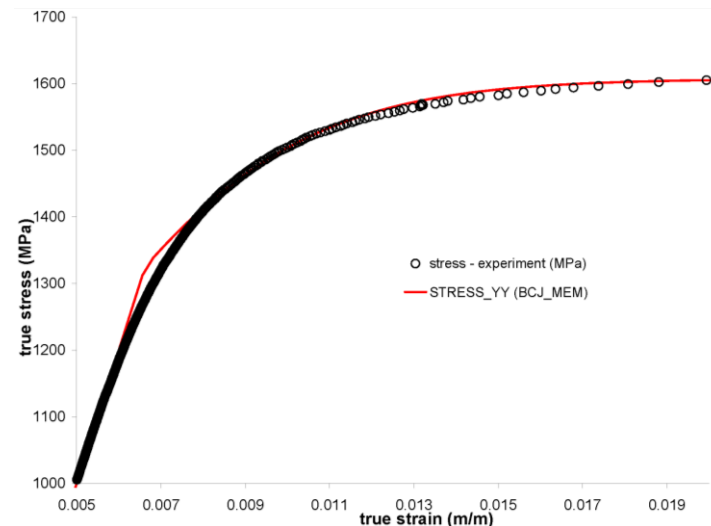
Supports Credible Prediction

- Dakota can help with parts of a V&V workflow, crucial for credible prediction
- Example: *ASME V&V 10 Guide for Verification and Validation in Computational Solid Mechanics*

sensitivity analysis to down-select

design of computer experiments

verification

calibration / comparison with data

uncertainty-aware validation

WHY USE DAKOTA?

Dakota: Distinguishing Strengths

- Makes sensitivity analysis, optimization, and uncertainty quantification practical for costly computational models

- Flexible interface to simulation codes: one interface; many methods

- Combined deterministic/probabilistic analysis

- Continual advanced algorithm R&D to tackle computational challenges (particularly in SNL's national security mission)
  - Treats non-smooth, discontinuous, multi-modal responses
  - Surrogate-based, multi-fidelity, and hybrid methods
  - Risk-informed decision-making: epistemic and mixed UQ, rare events, Bayesian

- Scalable parallel computing from desktop to HPC

Many Methods in One Tool

| Sensitivity Analysis	Uncertainty Quantification
• Designs: MC/LHS, DACE, sparse grid, one-at-a-time • Analysis: correlations, scatter, Morris effects, Sobol indices	• MC/LHS/Adaptive Sampling • Reliability • Stochastic expansions • Epistemic methods
Optimization	Calibration
• Gradient-based local • Derivative-free local • Global/heuristics • Surrogate-based	• Tailored gradient-based • Use any optimizer • Bayesian inference

Interface Dakota to your simulation once, then apply various algorithms depending on your goal…

# Engineering Needs Drive Dakota R&D

Advanced approaches help you solve practical problems:

- Characterize parameter uncertainty → Bayesian calibration

- Hybrid analysis → mix methods, surrogates, and models

- Mixed uncertainty characterizations →
epistemic and mixed UQ approaches

- Costly simulations → surrogate-based optimization and UQ

- Build in safety or robustness →
mixed deterministic/ probabilistic methods
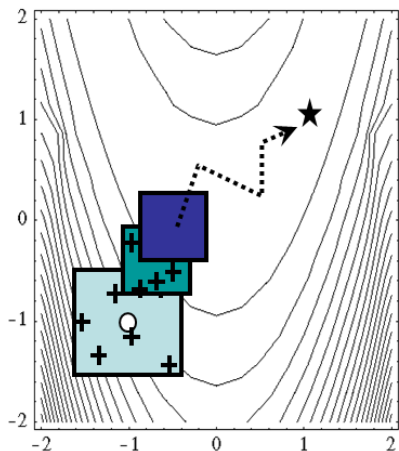
$$\min \quad f(d) + W s_u(d)$$
$$\text{s.t.} \quad g_l \leq g(d) \leq g_u$$
$$h(d) = h_t$$
$$d_l \leq d \leq d_u$$
$$a_l \leq A_i \, s_u(d) \leq a_u$$
$$A_e \, s_u(d) = a_t$$

bound probability
or bound response
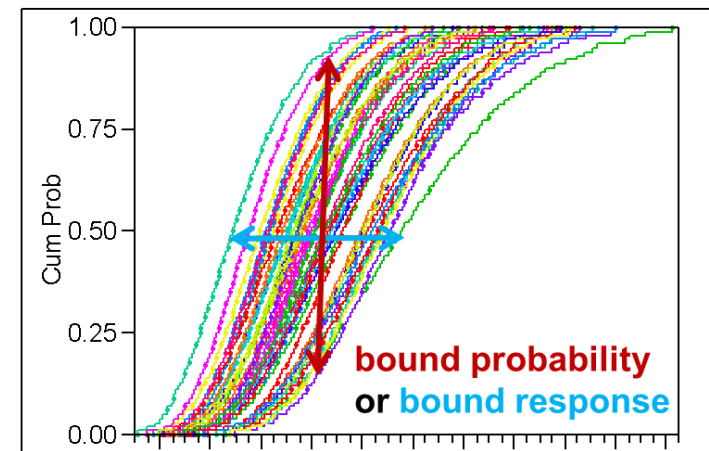
Computing and Parallelism

- Runs in various computing environments
  - Desktop: Mac, Linux, Windows
  - HPC: Linux clusters, IBM Blue Gene/P and /Q, IBM AIX, including many DOE machines
  - Distributed workstation computing
- Exploits concurrency at multiple levels
  - Multiprocessor simulations
  - Multiple simulations per response
  - Samples in a parameter study
  - Optimizations from multiple starting points
- File management features, including
  - Work directories to partition analysis files
  - Template directories share files common among analyses
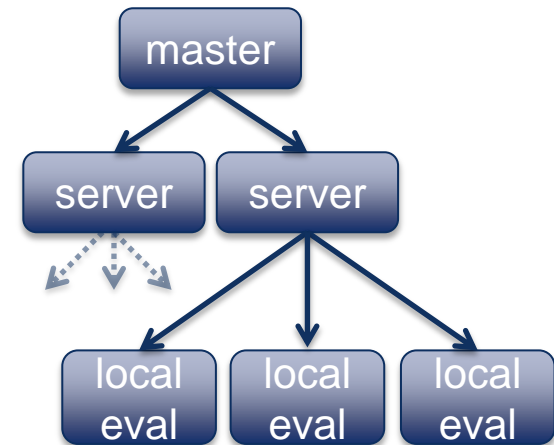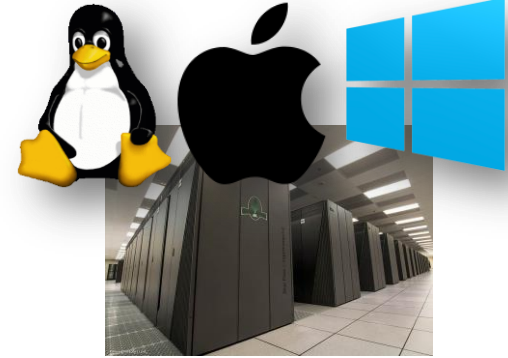
Dakota History and Resources

- Genesis: 1994 optimization LDRD

- Modern software quality and development practices

- Released every May 15 and Nov 15

- Established support process for SNL, partners, and beyond

Mike Eldred, Founder

http://dakota.sandia.gov

Lab mission-driven algorithm R&D deployed in production software

- Extensive website: documentation, training materials, downloads

- Open source facilitates external collaboration; widely downloaded

PREREQUISITES FOR USING DAKOTA

Intended Audience

- Primarily used by computational scientists and engineers, who work with simulations/models

Helpful background:

- Familiarity with mathematics, statistics, computer science
- Scripting or programming to create a Dakota-to-simulation interface
- Comfort with text-based input files and command-line interface
- Familiarity with plotting or visualization tools to post-process Dakota results

What Simulations Work with Dakota?

- **Applied to many science and engineering domains:** mechanics, structures, shock, fluids, electrical, radiation, bio, chemistry, climate, infrastructure, etc.

- **Example simulation codes:** finite element, discrete event, Matlab, Python models

- Helpful simulation characteristics:
  - Can be run in a non-interactive / batch mode
  - Parameters (inputs) not hard-wired, can be adjusted
  - Simulation responses (outputs) can be programmatically processed to extract a few key quantities of interest
  - Model is robust to parameter variations

radiation transport

molecular dynamics

electrical circuits
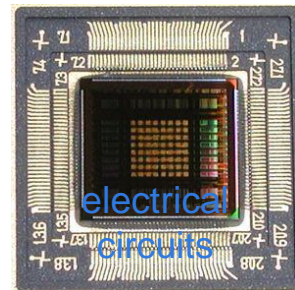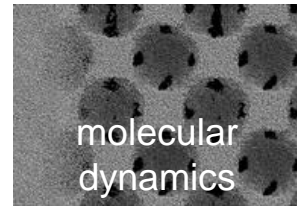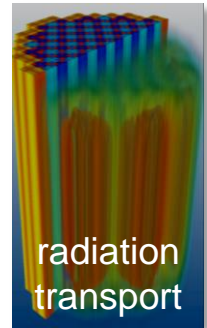
Getting Started and Getting Help

Tour http://dakota.sandia.gov at a high level

- **Getting Started**
  - Download (LGPL license, freely available worldwide): http://dakota.sandia.gov/download.html
  - Getting Started: http://dakota.sandia.gov/quickstart.html
  - User's Manual, Chapter 2: Tutorial with example input files http://dakota.sandia.gov/sites/default/files/docs/6.2/Users-6.2.0.pdf

- **Getting Help**
  - Extensive documentation (user, reference, developer): http://dakota.sandia.gov/content/manuals
  - Support mailing list (reaches Dakota team and user community): dakota-users@software.sandia.gov